

Problems in developing a comprehensive HL7 Database

Frank Oemig, Dipl.Inf.¹⁾, Joachim Dudeck, M.D.,¹⁾²⁾

German HL7 User Group¹⁾

Institut for Medical Informatics, University of Giessen²⁾

The German HL7 user group has edited a German adaptation of the version 2.1 and 2.2 of the HL7 standard. The careful analysis of the standard during the translation process and the attempt to map the standard into a comprehensive database has displayed some problems which should be taken into consideration in future developments in particular of version 3.x..

Within Hospital Information Systems (HIS) patient data, orders result reports, queries etc. have to be transmitted continuously. Communication standards are to diminish the efforts required to connect different hospital computer systems. The standard most often used within HIS is now Health Level Seven (HL7) which is also frequently applied in Europe. HL7 has recently been approved as an official ANSI standard. The HL7 user group in Germany has adapted the standard to the German environment and has tried to map the standard into a comprehensive database. In this process two important problems have been detected which should be taken into consideration in further versions of the standard.

The general paradigm of HL7 is the exchange of messages which consist of sequences of mandatory or optional segments, fields, components and subcomponents. Messages are initiated by trigger events, i.e. real world event which require the transmission of a specific data set. In the current versions of HL7 messages are characterized by message and event types which are transmitted as components of item 9 in the message header. But these two important descriptors of the message are unfortunately not unequivocally used within the standard. The message type describes sometimes the domain the message belongs to like ADT, sometimes the structure and the purpose of the message which is e.g. in the ADT domain defined by the event type. Each message requires a certain type of acknowledgment which is in Version 2.3 allocated to the event type.

For improving the consistency of the standard we recommend to characterize the message by four components,

- the domain the message belongs to,
- the structure of the message i.e. the sequence of segments and the contents of fields,
- the purpose of the message and
- the expected type of acknowledgment.

Using these components for describing messages would decrease the number of messages to be defined in the standard, diminish implementation problems and would in particular make the standard more consistent.

Our main goal was the mapping of the complete standard into a comprehensive HL7 database. Using the described parameters for characterizing messages the relationship between messages, trigger events, messages types, segments and one component fields can easily be represented in the HL7 database. Problems occurred in the mapping process when HL7 attribute tables are allocated not to fields but to components. Since version 2.2 more and more multiple component fields are introduced. We became aware that two different types of multiple component fields have to be distinguished, one with independent and others with dependent components which have to be handled distinctly in the mapping process. In fields with independent components attribute tables allocated to components can be mapped into the HL7 database if the components are handled in the same way as data items in the standard. But up to now components cannot be entered into the data element table.

Other Problems occurred in data items with dependent components like in the data type 'CE' (Coded element) where attribute tables allocated to components are dependent on other components. CE consist e.g. of three components

identifier^text^name of coding system

Attribute Tables of identifiers are dependent on the chosen coding system. We have solved the occurring problems by introducing into the design of the HL7 database an additional object, the 'logical data type' (LDT). The logical data type represent the multiple component field as atomic information object handled in the mapping process in the same way as single component data items.